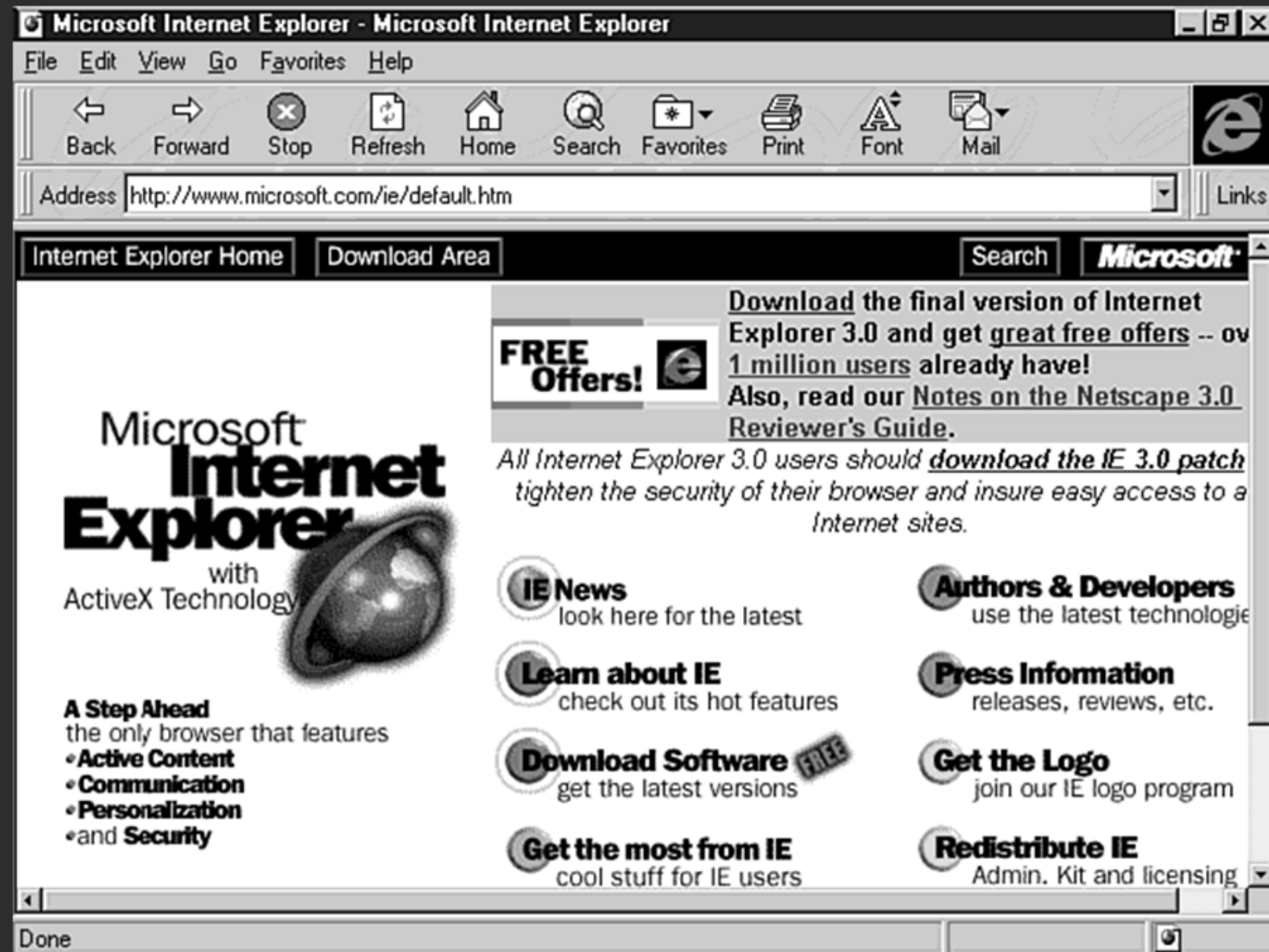


# Where does CSS come from?

Rachel Andrew CSSConf EU 2017

1996



Don't break  
the web!



Where does CSS  
come from **today**?







W3 CSS WG members

www.w3.org/Style/CSS/members

Search Yahoo!

MEMBERS

The following is the complete list of members & [W3C member organizations](#) that participate in the CSS working group. (See [“How to become a W3C member”](#) and the [Participation FAQ](#) for information on joining W3C as a member.)

Name	Affiliation
Glenn Adams.	Skynav, Inc.
Rachel Andrew	W3C Invited Experts
Rossen Atanassov	Microsoft Corporation
Tab Atkins Jr.	Google, Inc.
Takao Baba	Beyond Perspective Solutions ltd
Shezan Baig	Bloomberg
Mihail-Alexandru Balan	Adobe
David Baron.	Mozilla Foundation
Brian Birtles.	Mozilla Foundation
Bert Bos.	W3C Staff
Bogdan Brinza	Microsoft Corporation
Rob Buis	Samsung Electronics Co., Ltd.
Rick Byers	Google, Inc.
Bo Campbell	IBM Corporation
Angelo Cano	Antenna House, Inc.
Adenilson Cavalcanti.	Samsung Electronics Co., Ltd.
Tantek Çelik	Mozilla Foundation
Junichi Chiba	Beyond Perspective Solutions ltd
Dave Cramer.	Hachette Livre
Alex Critchfield.	Antenna House, Inc.
John Daggett	W3C Invited Experts
Beth Dakin.	Apple, Inc.
Alex Danilo	Google, Inc.
Greg Davis.	Pearson plc
Benjamin De Cock	Stripe
Alexis Menard.	Intel Corporation
Thierry Michel.	W3C Staff
Michael Miller	Antenna House, Inc.
Shinyu Murakami.	Vivliostyle Inc.
Rachel Nabors.	Microsoft Corporation
Masayuki Nakano.	Mozilla Foundation
Ryosuke Niwa.	Apple, Inc.
Theresa O'Connor	Apple, Inc.
David O'Dey	Pearson plc
Christopher Palmer.	Bloomberg
Simon Pieters.	Opera Software AS
Benjamin Poulain.	Apple, Inc.
Deblyn prado.	NIC.br - Brazilian Network Information Center
Anton Prowse.	W3C Invited Experts
Liam Quin	W3C Staff
Matt Rakow.	Microsoft Corporation
François Remy.	Microsoft Corporation
Melanie Richards.	Microsoft Corporation
Florian Rivoal.	Vivliostyle Inc.
Philip Rogers	Google, Inc.
Jacob Rossi.	Microsoft Corporation
Andrey Rybka	Bloomberg
Hiroshi Sakakibara.	Beyond Perspective Solutions ltd
Simon Sapin.	Mozilla Foundation
Nathalia Sautchuk Patrício	NIC.br - Brazilian Network Information Center
Dirk Schulze	Adobe
Jen Simmons	Mozilla Foundation
David Singer.	Apple, Inc.
Geoffrey Sneddon.	W3C Invited Experts
Hyojin Song.	LG Electronics
Alan Stearns.	Adobe
Shane Stephens	Google, Inc.

# Specs originate from ...

- ☞ Browser vendors
- ☞ Other CSS User Agents (UAs for print, EPUB etc.)
- ☞ Companies such as Adobe
- ☞ From the CSS Working Group itself

# CSS Grid

A new spec is born

# 7 April 2011

## First Public Working Draft<sup>1</sup>

---

<sup>1</sup> CSS Grid Layout, First Public Working Draft

# CSS Advanced Layout Module

# 15 December

# 2005<sup>2</sup>

---

<sup>2</sup> <https://www.w3.org/TR/2005/WD-css3-layout-20051215/>

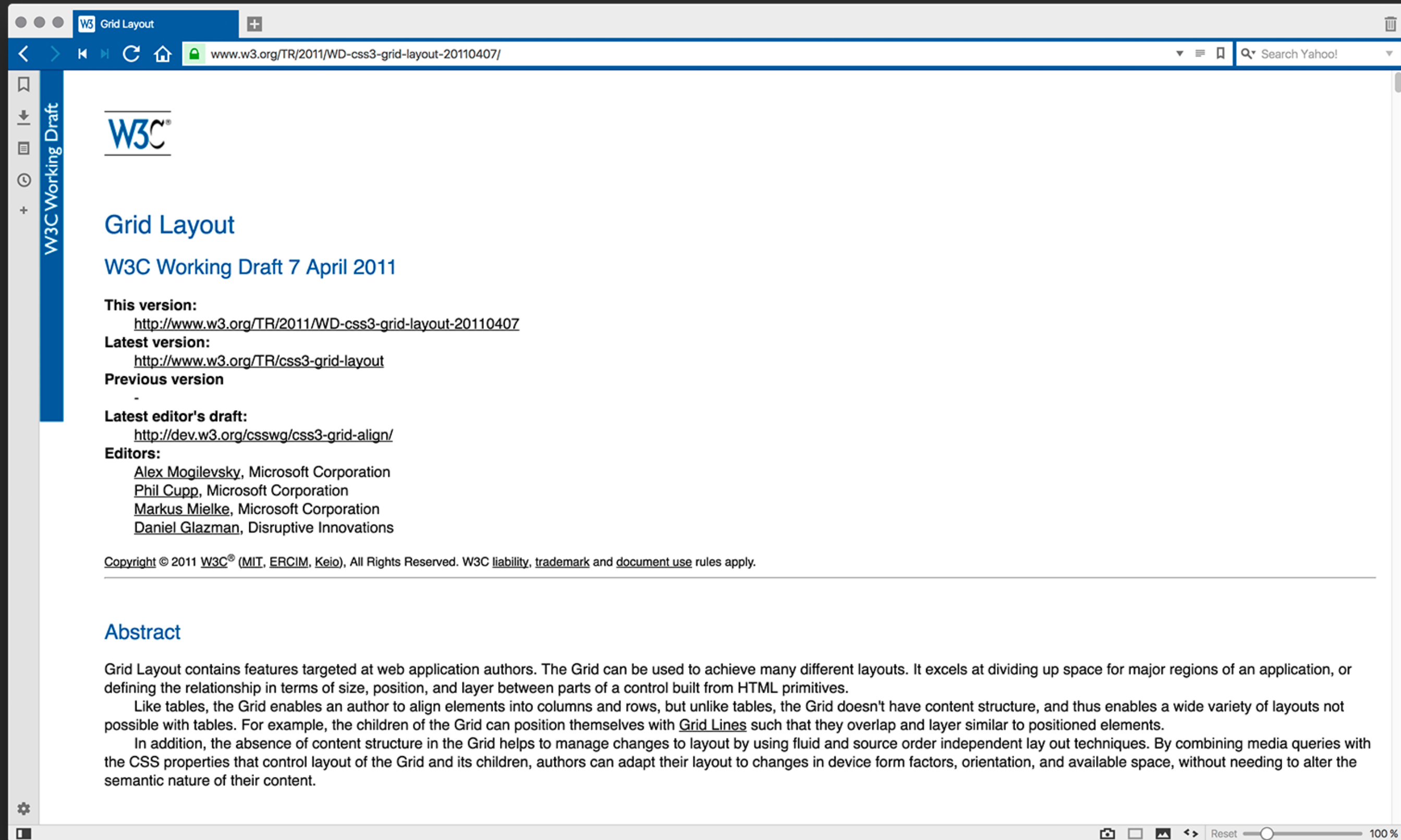
# A Working Draft

A snapshot of the state of a specification at a particular date.



# The Editor's Draft

The current document in progress. This may be significantly different to the most recent Working Draft.



# Experimental implementations

*If you like CSS these are fun to play with!*

The screenshot shows a web browser window with a red address bar. The address bar contains the URL `24ways.org/2012/css3-grid-layout/` and a search bar with the text "Search Yahoo!". The page title is "Creating grid systems". The main content area has a light blue background. On the left side, there is a vertical sidebar with icons for bookmark, download, list, clock, and a plus sign. On the right side, there is a vertical sidebar with a hamburger menu icon, left and right arrow icons, and a scroll bar. The main text area contains the following content:

## Creating grid systems

As soon as I started to play with CSS3 grid layout, I wanted to see if I could use it to replicate a flexible grid system like this [fluid 16-column 960 grid system](#).

I started out by defining a grid on my wrapper element, using fractions to make this grid fluid.

```
.wrapper {
  width: 90%;
  margin: 0 auto 0 auto;
  display: -ms-grid;
  -ms-grid-columns: 1fr (4.25fr 1fr)[16];
  -ms-grid-rows: (auto 20px)[24];
}
```

Like the 960 grid system I was using as an example, my grid starts with a gutter, followed by the first actual column, plus another gutter repeated sixteen times. What this means is that if I want to span two columns, as far as the grid layout module is concerned that is actually three columns: two wide columns, plus one gutter. So this needs to be accounted for when positioning items.

I created a CSS class for each positioning option: column position; rows position; and column span. For example:

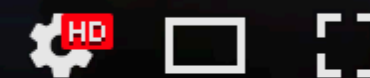
```
.grid1 {-ms-grid-column: 2;} /* applying this class positions an item in the 2nd column
.grid2 {-ms-grid-column: 4;} /* 2nd column - gutter|column 1|gutter
.grid3 {-ms-grid-column: 6;} /* 3rd column - gutter|column 1|gutter|column 2|gutter
```

The bottom of the browser window shows a status bar with icons for print, full screen, and a zoom slider set to 100%.





0:13 / 28:56



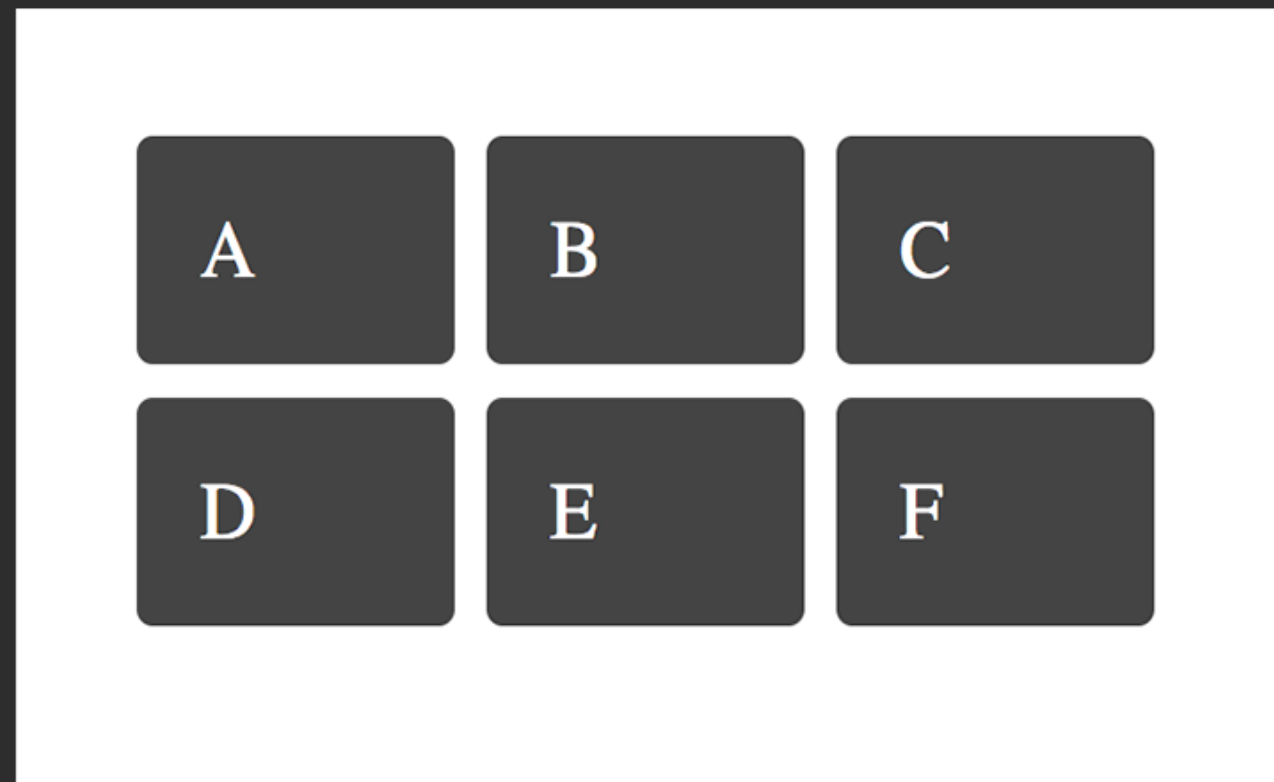
Rachel Andrew | @rachelandrew | Slides & Resources <https://rachelandrew.co.uk/speaking/event/cssconfeu-2017>

“A great deal of the content delivered on the Bloomberg Terminal is actually rendered using web technologies such as JavaScript, HTML and CSS [...] For years, we have sponsored Igalia to implement and land advancements in the Chromium project and its JavaScript engine, V8, and participated in the standards-creation processes that define how these tools will operate in the future.”

👉 <https://www.techatbloomberg.com/blog/bloombergs-2016-open-source-contributions-5-top-projects/>

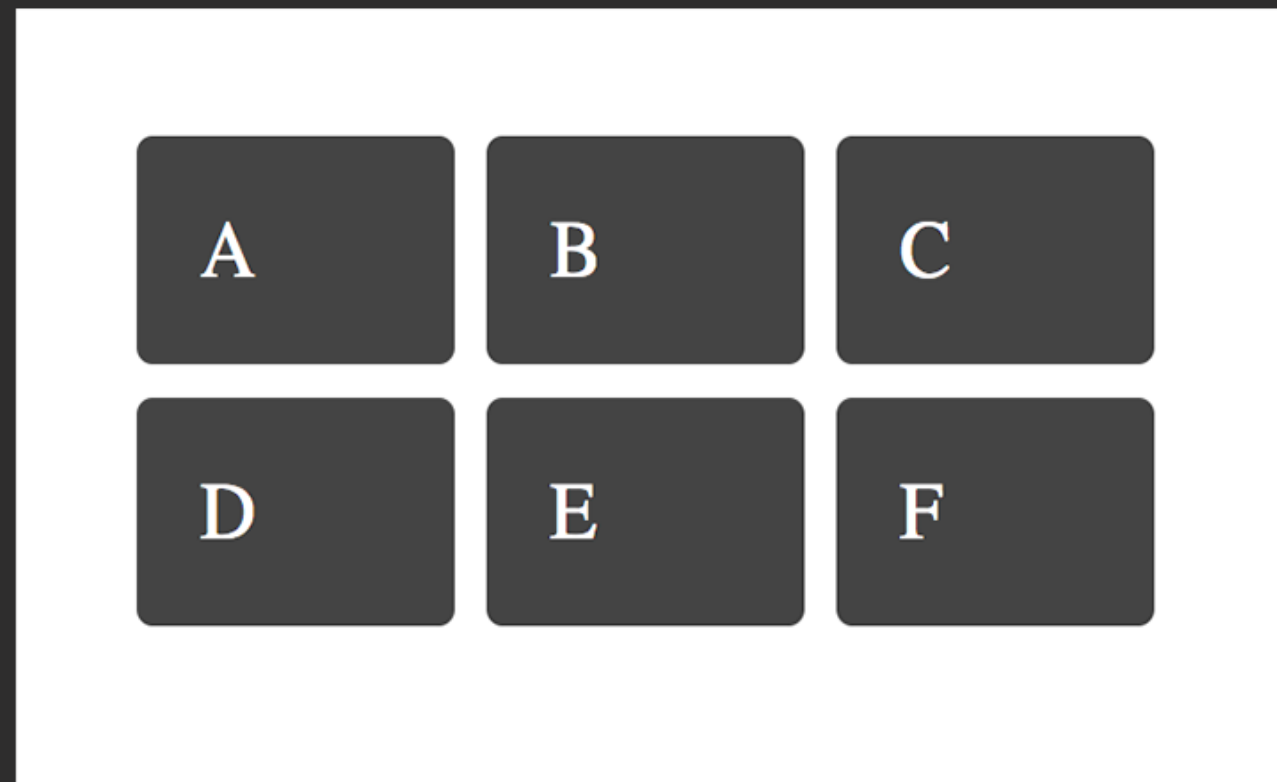
# The **gutter** problem

```
.grid {  
  display: grid;  
  grid-template-columns: 100px 10px 100px 10px 100px;  
  grid-template-rows: auto 10px auto;  
}
```





```
.grid {  
  display: grid;  
  grid-template-columns: 100px 100px 100px;  
  grid-gap: 10px;  
}
```



By using experimental implementations,  
the web design and development community  
**can influence specs**

Browsers implement  
alongside spec development

# Browser flags vs. Vendor prefixes

- ☞ **Flexbox**: experimental implementation shipped using vendor prefixes.
- ☞ **CSS Grid**: experimental implementation behind browser flags

# CSS Grid Candidate Recommendation

29 September 2016<sup>3</sup>

---

<sup>3</sup> <https://www.w3.org/TR/2016/CR-css-grid-1-20160929/>

# Candidate Recommendations

“A Candidate Recommendation is a document that W3C believes has been widely reviewed and satisfies the Working Group’s technical requirements. W3C publishes a Candidate Recommendation to gather implementation experience.”<sup>4</sup>

---

<sup>4</sup> <https://www.w3.org/2005/10/Process-20051014/tr#maturity-levels>

# Next step: Proposed Recommendation

Requires two independent interoperable implementations of each feature

# at-risk features



# March 2017

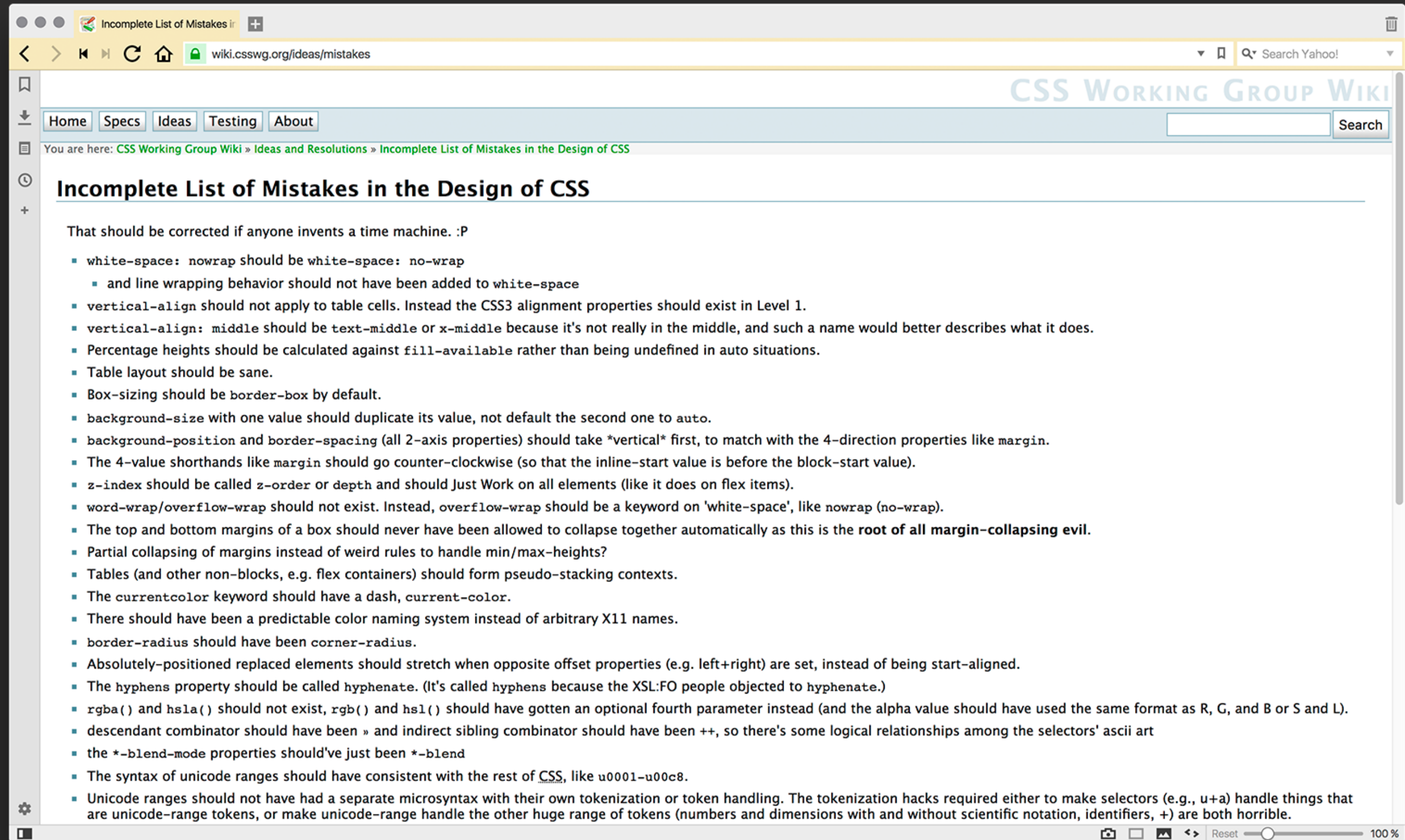


Can this be  
**implemented?**

What **priority** are  
implementors giving it?

Does this make  
sense to **authors**?

Will this  
**break** anything?



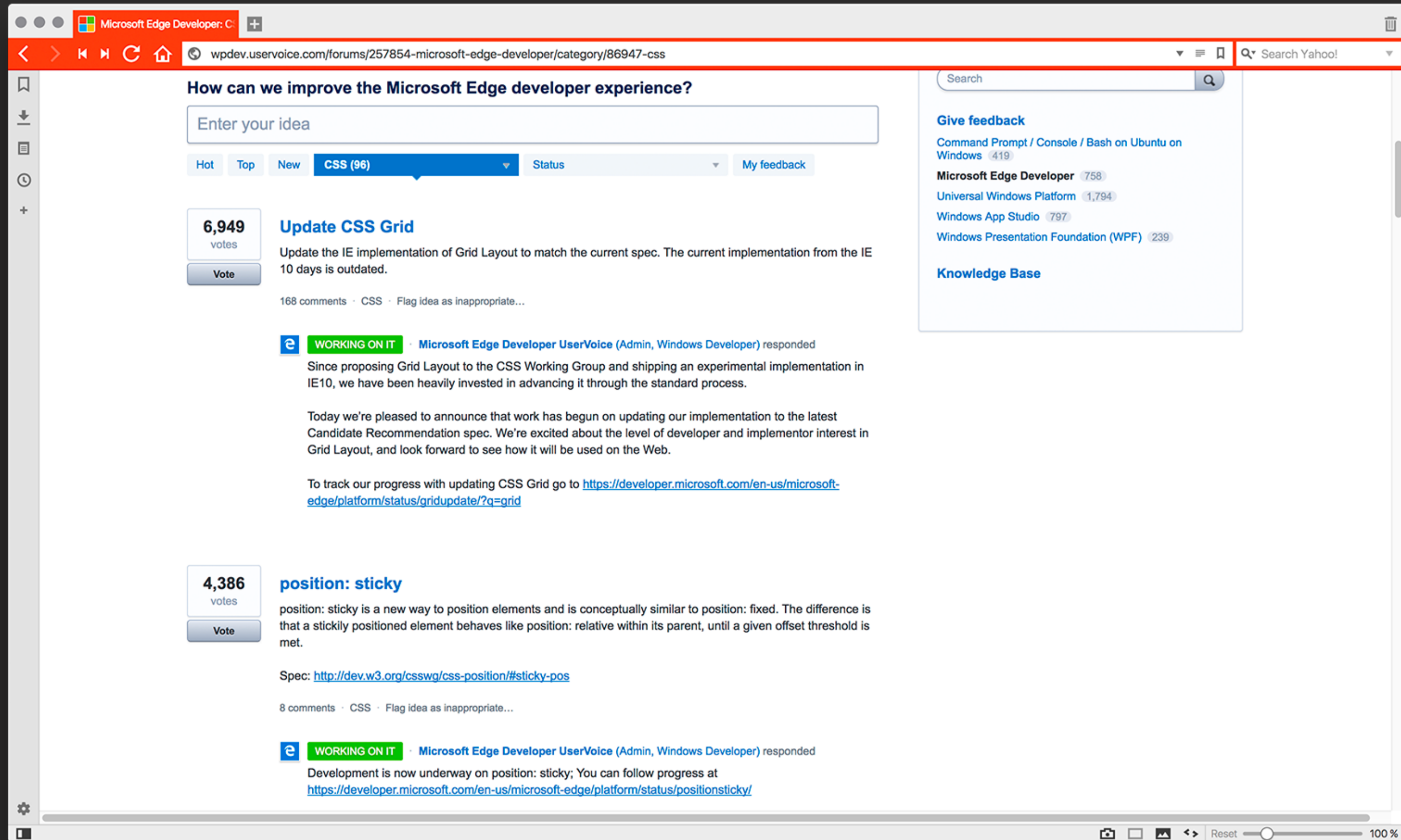
# If some CSS seems **weird**

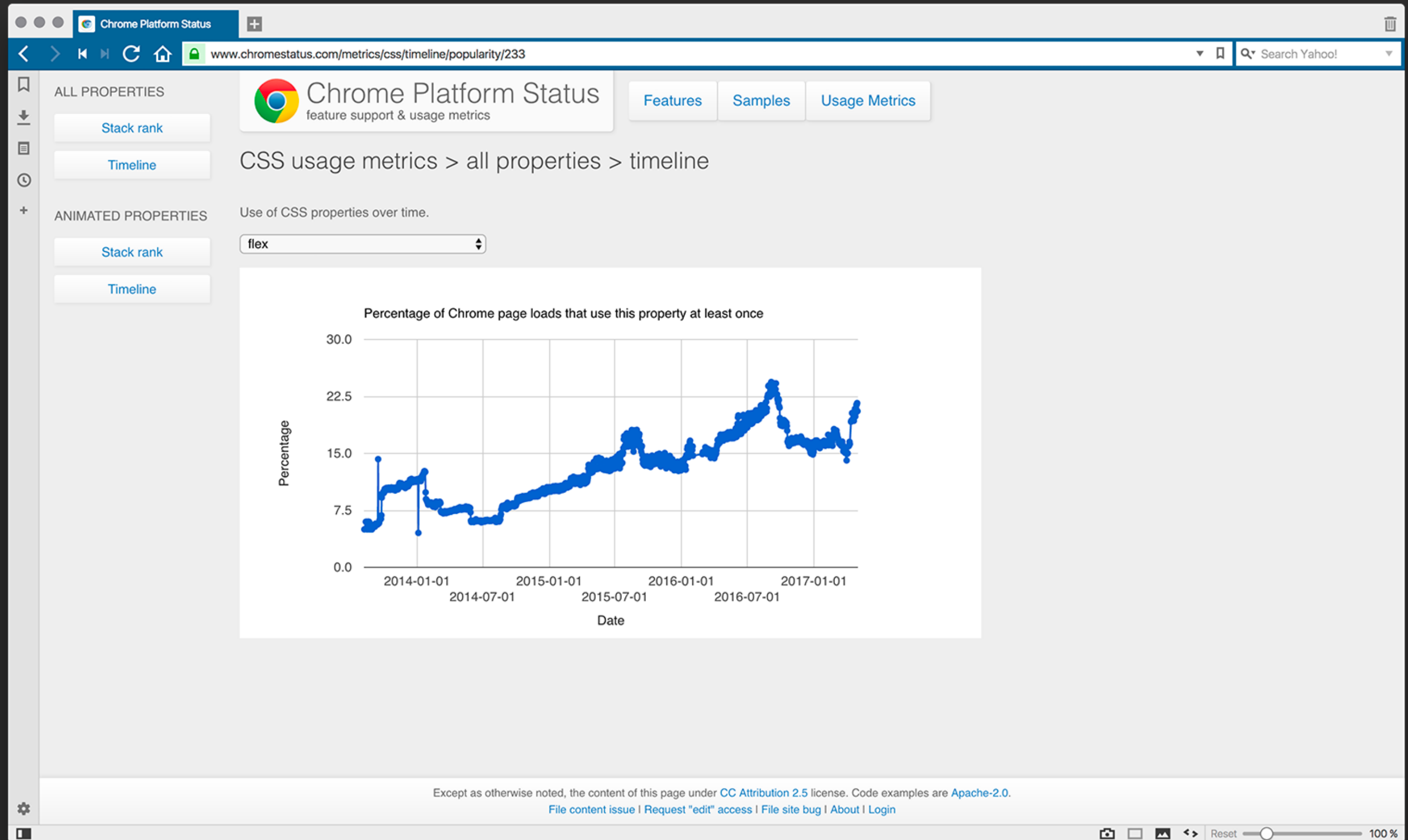
Doing a bit of digging and finding out **why** it is that way can help your understanding of that property.

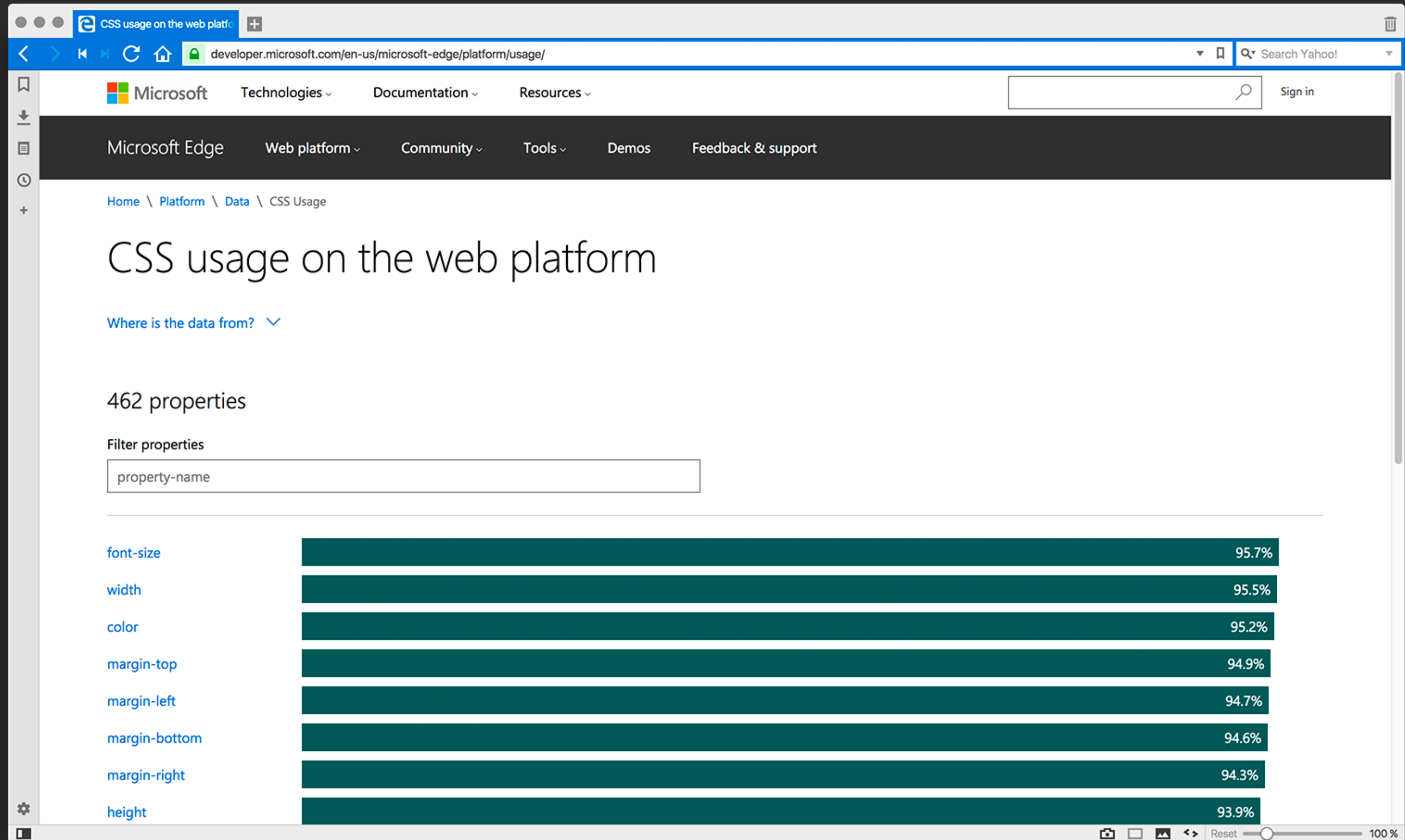
# You can help to influence the direction of CSS

- ① Ask for features to be implemented in browsers
- ② Use and write about new browser features
- ③ Share your ideas and feedback with the CSS Working Group
- ④ Log bugs with specifications and with browsers









I will ping you when we begin working on updating our implementation.

*Closing sentence from Microsoft to Rachel Andrew in response to her regarding Grid in 2014*

# Take part in developing the **web platform**

Share your ideas and feedback with the CSS Working Group

# The CSS Working Group

- 👉 GitHub: <https://github.com/w3c/csswg-drafts>
- 👉 Twitter: <https://twitter.com/csswg>
- 👉 Current Work: <https://www.w3.org/Style/CSS/current-work.en.html>

# Show your use case when suggesting a feature

- ☞ what problem will this feature solve?

- ☞ are authors already working around this in some way?

- ☞ why are those existing methods problematic?



css-grid

Decorative grid-cell

github.com/w3c/csswg-drafts/issues/499

Search Yahoo!

Features

Business

Explore

Pricing

This repository

Search

Sign in

Sign up

w3c / csswg-drafts

Watch

160

Star

472

Fork

112

Code

Issues 618

Pull requests 16

Projects 3

Pulse

Graphs

[css-grid] Decorative grid-cell pseudo-elements #499

New issue

Open

tabatkins opened this issue on Sep 19 2016 · 6 comments

X

tabatkins commented on Sep 19 2016

Member

@jensimmons

 proposes adding a grid-cell pseudo so you can add (responsive) decorative elements to grids without having to add empty elements to your page.

Syntax like:

```
#grid::grid-area(1 / 2 / 3 / 4) {
  background-color: red; /* etc */
  /* grid-positioning properties are blacklisted */
}
```

Also some way of controlling whether or not that takes up auto-flow space. Might want to reserve space or not. Can we base this on 'content' or something? Or do we need a new property?

👍 3

X

tabatkins added the `css-grid-2` label on Sep 19 2016

jensimmons commented on Sep 19 2016

The slides from the presentation I just gave are at: <https://speakerdeck.com/jensimmons/proposal->

Assignees

No one assigned

Labels

`css-grid-2`

Projects


None yet


Milestone


No milestone


7 participants


X

















Rachel Andrew | @rachelandrew | Slides & Resources <https://rachelandrew.co.uk/speaking/event/cssconfeu-2017>

# Log **spec** issues

CSS specifications have bugs too! Raise those as Issues on the CSS WG GitHub at [github.com/w3c/csswg-drafts/issues/](https://github.com/w3c/csswg-drafts/issues/).

# Raise issues with **browsers**

Log any interoperability issues you see between browsers.

👉 Edge: <https://developer.microsoft.com/en-us/microsoft-edge/platform/issues/>

👉 Firefox: <https://bugzilla.mozilla.org/>

👉 Chrome: <https://bugs.chromium.org/p/chromium/issues/list>

👉 Safari: <https://bugs.webkit.org/buglist.cgi?component=CSS&product=WebKit>

# Your dev superpower

Creating a **Reduced Test Case**.

👉 <https://rachelandrew.co.uk/archives/2017/01/30/reporting-browser-bugs/>

👉 [https://developer.mozilla.org/en-US/docs/Mozilla/QA/Reducing\\_testcases](https://developer.mozilla.org/en-US/docs/Mozilla/QA/Reducing_testcases)

👉 <https://webkit.org/test-case-reduction/>

# **Contributing to CSS**

is an amazing way to

**learn** about CSS

Where does  
**CSS** come from?







# Thank **you!**

<https://rachelandrew.co.uk/speaking/event/cssconfeu-2017>

## **Find me at:**

👉 <https://rachelandrew.co.uk>

👉 <https://grabaperch.com>

👉 <http://gridbyexample.com>

👉 <https://twitter.com/rachelandrew>